Carbon Dioxide Emissions

Josh Browning

18/02/2015

## Read in a csv file  
setwd("~/Documents/Projects/Ad Hoc/Rocio\_Condor/") # Set your directory  
emissionsData = read.csv(file = "italyData.csv")  
  
## Look at the data  
dim(emissionsData)

## [1] 24 6

head(emissionsData)

## AreaName ElementName ItemName Year Value Flag  
## 1 Italy Emissions (CO2eq) Agriculture total 1990 39859.32 A  
## 2 Italy Emissions (CO2eq) Agriculture total 1991 38173.47 A  
## 3 Italy Emissions (CO2eq) Agriculture total 1992 37629.37 A  
## 4 Italy Emissions (CO2eq) Agriculture total 1993 36513.03 A  
## 5 Italy Emissions (CO2eq) Agriculture total 1994 36017.06 A  
## 6 Italy Emissions (CO2eq) Agriculture total 1995 35138.63 A

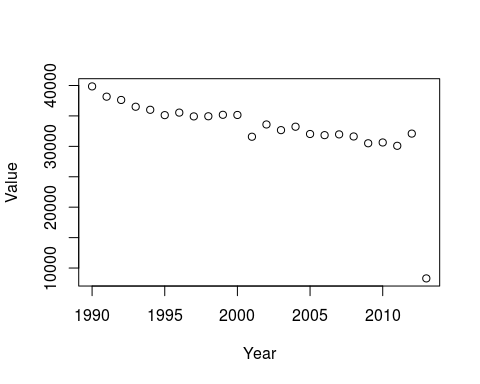
tail(emissionsData)

## AreaName ElementName ItemName Year Value Flag  
## 19 Italy Emissions (CO2eq) Agriculture total 2008 31627.90 A  
## 20 Italy Emissions (CO2eq) Agriculture total 2009 30511.02 A  
## 21 Italy Emissions (CO2eq) Agriculture total 2010 30639.55 A  
## 22 Italy Emissions (CO2eq) Agriculture total 2011 30094.77 A  
## 23 Italy Emissions (CO2eq) Agriculture total 2012 32100.87 A  
## 24 Italy Emissions (CO2eq) Agriculture total 2013 8300.54 A

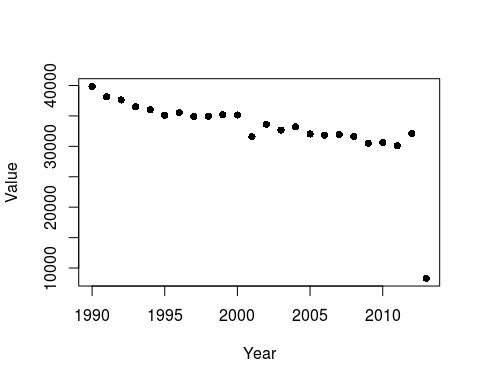
str(emissionsData)

## 'data.frame': 24 obs. of 6 variables:  
## $ AreaName : Factor w/ 1 level "Italy": 1 1 1 1 1 1 1 1 1 1 ...  
## $ ElementName: Factor w/ 1 level "Emissions (CO2eq)": 1 1 1 1 1 1 1 1 1 1 ...  
## $ ItemName : Factor w/ 1 level "Agriculture total": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Year : int 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 ...  
## $ Value : num 39859 38173 37629 36513 36017 ...  
## $ Flag : Factor w/ 1 level "A": 1 1 1 1 1 1 1 1 1 1 ...

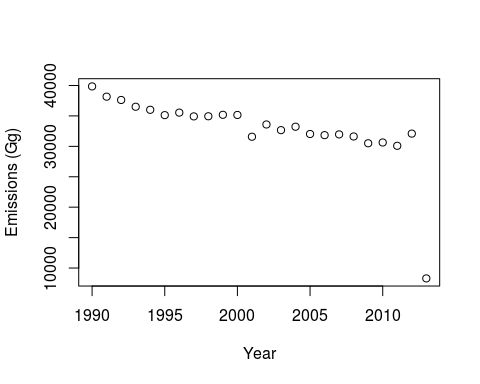
## Basic Plotting: One country over time  
plot(Value ~ Year, data = emissionsData)



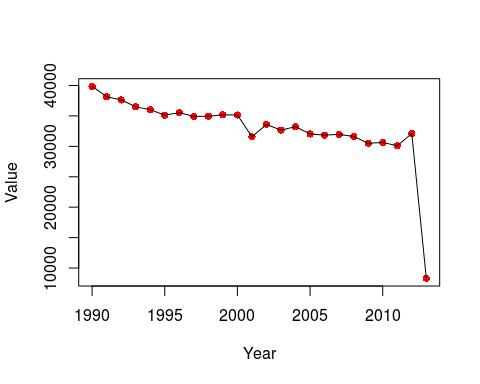
plot(Value ~ Year, data = emissionsData, pch = 16)



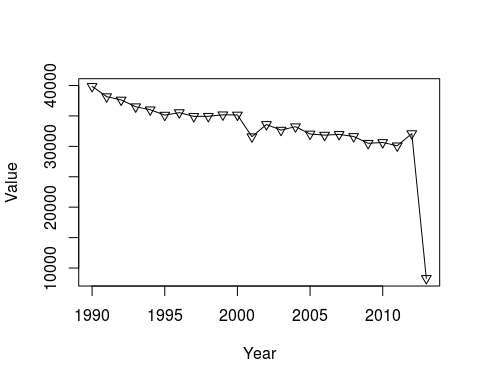
plot(Value ~ Year, data = emissionsData, ylab = "Emissions (Gg)")



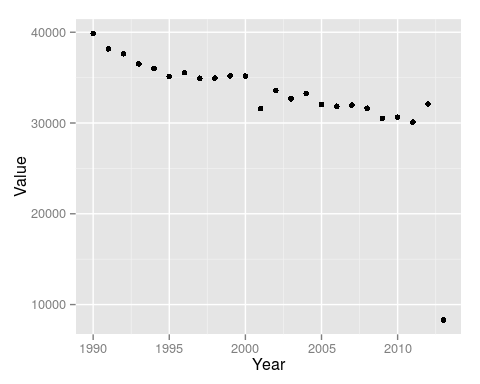
plot(Value ~ Year, data = emissionsData, col = 2, pch = 16)  
# lines() and points() add to the current plot  
lines(Value ~ Year, data = emissionsData)



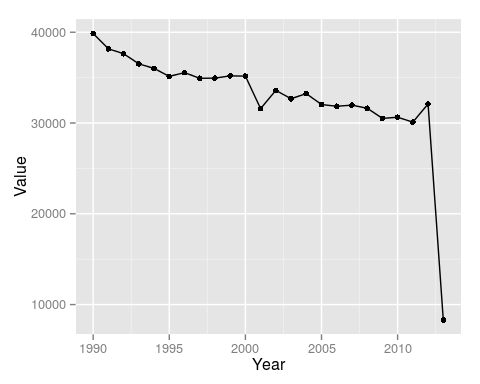
plot(Value ~ Year, data = emissionsData, type = "l") # "l" for "line"  
points(Value ~ Year, data = emissionsData, pch = 6)  
  
## More elegant plotting: use ggplot2  
library(ggplot2)



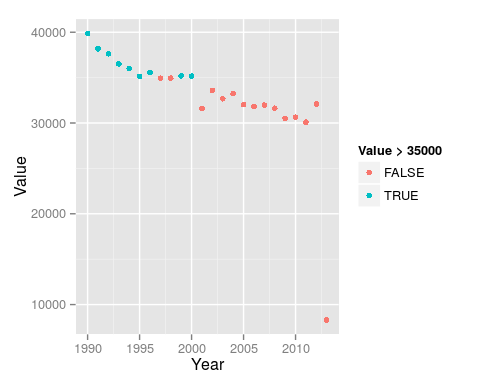
ggplot(emissionsData, aes(x = Year, y = Value)) +  
 geom\_point()



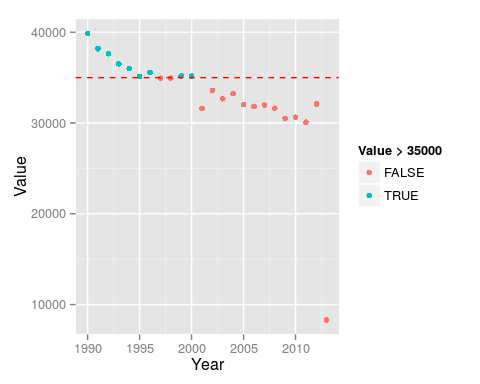
ggplot(emissionsData, aes(x = Year, y = Value)) +  
 geom\_point() + geom\_line()



ggplot(emissionsData, aes(x = Year, y = Value, color = Value > 35000)) +  
 geom\_point()



ggplot(emissionsData, aes(x = Year, y = Value, color = Value > 35000)) +  
 geom\_point() + geom\_hline(yintercept = 35000, color = "red", linetype = 2)



## Now, let's look at the full dataset from FAOSTAT  
emissionsData = read.csv(file = "faostat\_data.csv")  
dim(emissionsData)

## [1] 11620 12

head(emissionsData)

## Domain.Code Domain AreaCode AreaName ElementCode  
## 1 GT Agriculture Total 1 Armenia 7231  
## 2 GT Agriculture Total 1 Armenia 7231  
## 3 GT Agriculture Total 1 Armenia 7231  
## 4 GT Agriculture Total 1 Armenia 7231  
## 5 GT Agriculture Total 1 Armenia 7231  
## 6 GT Agriculture Total 1 Armenia 7231  
## ElementName ItemCode ItemName Year Value Flag  
## 1 Emissions (CO2eq) 1711 Agriculture total 1992 1380.60 A  
## 2 Emissions (CO2eq) 1711 Agriculture total 1993 1236.96 A  
## 3 Emissions (CO2eq) 1711 Agriculture total 1994 1164.76 A  
## 4 Emissions (CO2eq) 1711 Agriculture total 1995 1151.26 A  
## 5 Emissions (CO2eq) 1711 Agriculture total 1996 1163.91 A  
## 6 Emissions (CO2eq) 1711 Agriculture total 1997 1145.35 A  
## FlagD  
## 1 Aggregate, may include official, semi-official, estimated or calculated data  
## 2 Aggregate, may include official, semi-official, estimated or calculated data  
## 3 Aggregate, may include official, semi-official, estimated or calculated data  
## 4 Aggregate, may include official, semi-official, estimated or calculated data  
## 5 Aggregate, may include official, semi-official, estimated or calculated data  
## 6 Aggregate, may include official, semi-official, estimated or calculated data

tail(emissionsData)

## Domain.Code Domain  
## 11615 GT Agriculture Total  
## 11616 GT Agriculture Total  
## 11617 GT Agriculture Total  
## 11618 GT Agriculture Total  
## 11619   
## 11620 FAOSTAT Date: Fri Feb 13 15:57:17 CET 2015   
## AreaCode AreaName ElementCode  
## 11615 299 Occupied Palestinian Territory 7231  
## 11616 299 Occupied Palestinian Territory 7231  
## 11617 299 Occupied Palestinian Territory 7231  
## 11618 299 Occupied Palestinian Territory 7231  
## 11619 NA NA  
## 11620 NA NA  
## ElementName ItemCode ItemName Year Value Flag  
## 11615 Emissions (CO2eq) 1711 Agriculture total 2012 284.21 A  
## 11616 Emissions (CO2eq) 1711 Agriculture total 2013 185.89 A  
## 11617 Emissions (CO2eq) 1711 Agriculture total 2030 383.18 A  
## 11618 Emissions (CO2eq) 1711 Agriculture total 2050 417.33 A  
## 11619 NA NA NA   
## 11620 NA NA NA   
## FlagD  
## 11615 Aggregate, may include official, semi-official, estimated or calculated data  
## 11616 Aggregate, may include official, semi-official, estimated or calculated data  
## 11617 Aggregate, may include official, semi-official, estimated or calculated data  
## 11618 Aggregate, may include official, semi-official, estimated or calculated data  
## 11619   
## 11620

str(emissionsData)

## 'data.frame': 11620 obs. of 12 variables:  
## $ Domain.Code: Factor w/ 3 levels "","FAOSTAT Date: Fri Feb 13 15:57:17 CET 2015",..: 3 3 3 3 3 3 3 3 3 3 ...  
## $ Domain : Factor w/ 2 levels "","Agriculture Total": 2 2 2 2 2 2 2 2 2 2 ...  
## $ AreaCode : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ AreaName : Factor w/ 241 levels "","Afghanistan",..: 11 11 11 11 11 11 11 11 11 11 ...  
## $ ElementCode: int 7231 7231 7231 7231 7231 7231 7231 7231 7231 7231 ...  
## $ ElementName: Factor w/ 2 levels "","Emissions (CO2eq)": 2 2 2 2 2 2 2 2 2 2 ...  
## $ ItemCode : int 1711 1711 1711 1711 1711 1711 1711 1711 1711 1711 ...  
## $ ItemName : Factor w/ 2 levels "","Agriculture total": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Year : int 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 ...  
## $ Value : num 1381 1237 1165 1151 1164 ...  
## $ Flag : Factor w/ 2 levels "","A": 2 2 2 2 2 2 2 2 2 2 ...  
## $ FlagD : Factor w/ 2 levels "","Aggregate, may include official, semi-official, estimated or calculated data": 2 2 2 2 2 2 2 2 2 2 ...

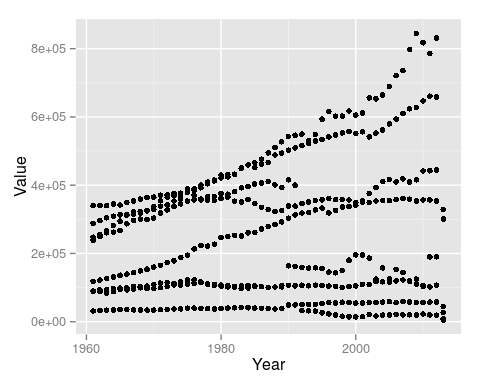
countrySubset = c("USSR", "Canada", "China, mainland",  
 "United States of America", "Brazil", "Australia", "India",  
 "Argentina", "Kazakhstan")  
# Now, let's only consider the 9 countries we sampled:  
library(dplyr)

##   
## Attaching package: 'dplyr'  
##   
## The following object is masked from 'package:stats':  
##   
## filter  
##   
## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

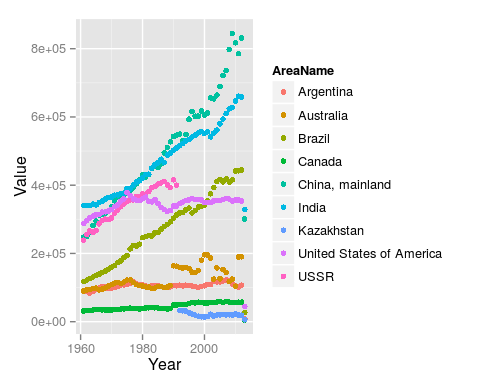
emissionsDataSubset = emissionsData %>%  
 filter(AreaName %in% countrySubset, Year <= 2015)  
dim(emissionsDataSubset)

## [1] 424 12

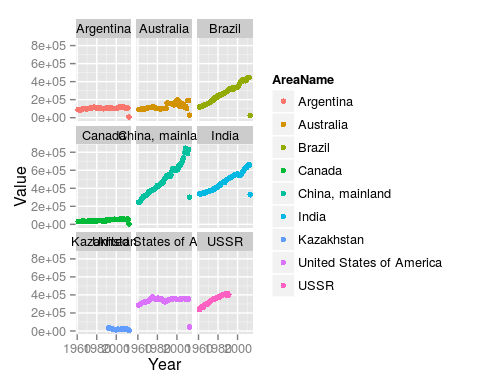
ggplot(emissionsDataSubset, aes(x = Year, y = Value)) +  
 geom\_point()



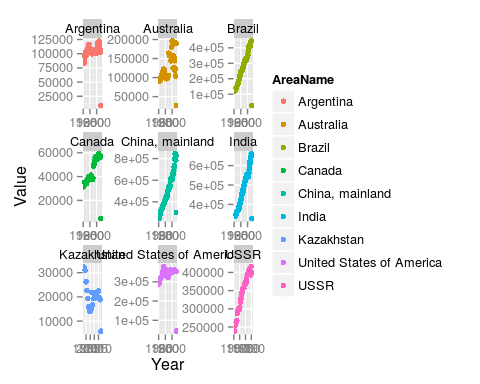
ggplot(emissionsDataSubset, aes(x = Year, y = Value, color = AreaName)) +  
 geom\_point()



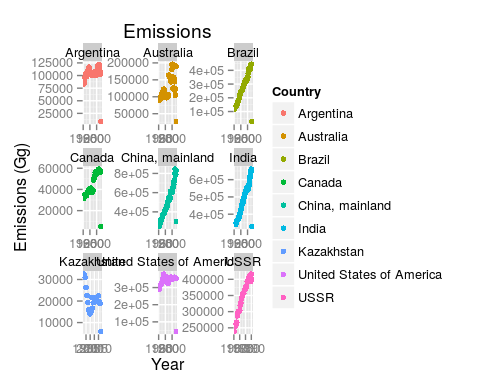
ggplot(emissionsDataSubset, aes(x = Year, y = Value, color = AreaName)) +  
 geom\_point() + facet\_wrap( ~ AreaName)



ggplot(emissionsDataSubset, aes(x = Year, y = Value, color = AreaName)) +  
 geom\_point() + facet\_wrap( ~ AreaName, scales = "free")

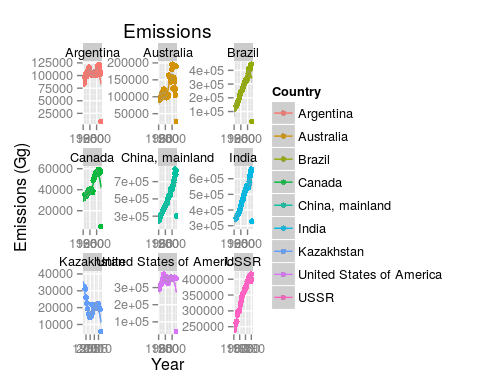


ggplot(emissionsDataSubset, aes(x = Year, y = Value, color = AreaName)) +  
 geom\_point() + facet\_wrap( ~ AreaName, scales = "free") +  
 labs(y = "Emissions (Gg)", color = "Country", title = "Emissions")

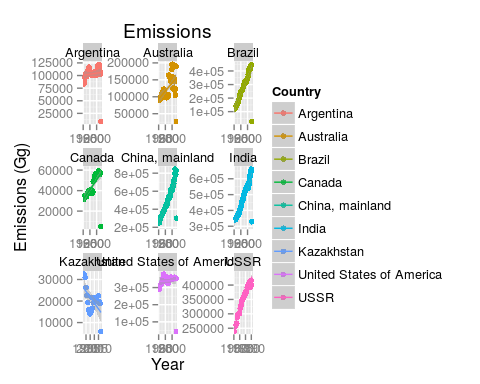


ggplot(emissionsDataSubset, aes(x = Year, y = Value, color = AreaName)) +  
 geom\_point() + facet\_wrap( ~ AreaName, scales = "free") +  
 labs(y = "Emissions (Gg)", color = "Country", title = "Emissions") +  
 geom\_smooth()

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.



ggplot(emissionsDataSubset, aes(x = Year, y = Value, color = AreaName)) +  
 geom\_point() + facet\_wrap( ~ AreaName, scales = "free") +  
 labs(y = "Emissions (Gg)", color = "Country", title = "Emissions") +  
 geom\_smooth(method = "lm")



## FAOSTAT let's us access this data directly:  
library(FAOSTAT)  
# FAOsearch() # 8 1 1 1 1  
# queriedData = getFAOtoSYB(query = .LastSearch)  
emissionsDataFAOSTAT = getFAO(  
 name = "Agriculture Total\_Agriculture total + (Total)\_Emissions (CO2eq)(NA)",  
 domainCode = "GT", itemCode = "1711", elementCode = "7231")

##   
## NOTE: Multiple China detected in 'Value' sanitization is performed

colnames(emissionsDataFAOSTAT)

## [1] "FAOST\_CODE"   
## [2] "Year"   
## [3] "Agriculture Total\_Agriculture total + (Total)\_Emissions (CO2eq)(NA)"

head(emissionsDataFAOSTAT)

## FAOST\_CODE Year  
## 1 2 1961  
## 2 3 1961  
## 3 4 1961  
## 4 5 1961  
## 5 7 1961  
## 6 8 1961  
## Agriculture Total\_Agriculture total + (Total)\_Emissions (CO2eq)(NA)  
## 1 9272.234651  
## 2 1794.475695  
## 3 2982.568450  
## 4 4.090834  
## 5 2091.974641  
## 6 15.263844

colnames(emissionsDataFAOSTAT)[3] = "Emissions"  
head(emissionsDataFAOSTAT)

## FAOST\_CODE Year Emissions  
## 1 2 1961 9272.234651  
## 2 3 1961 1794.475695  
## 3 4 1961 2982.568450  
## 4 5 1961 4.090834  
## 5 7 1961 2091.974641  
## 6 8 1961 15.263844

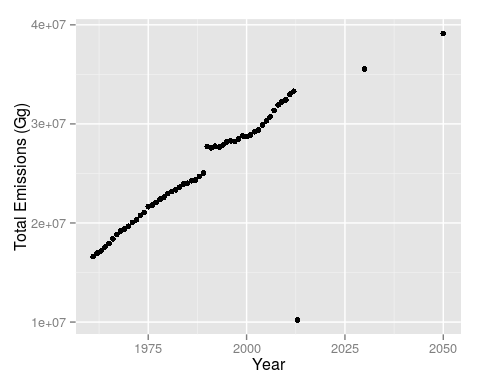
for(i in 1:10){  
 dataSubset = emissionsDataFAOSTAT %>%  
 filter(FAOST\_CODE == i)  
 fileName = paste0("plot\_", i, ".png")  
 png(fileName)  
 print(ggplot(dataSubset, aes(x = Year, y = Emissions)) +  
 geom\_point() + geom\_smooth() + labs(y = "Emissions (Gg)"))  
 dev.off() # Turn off plotting  
}

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.

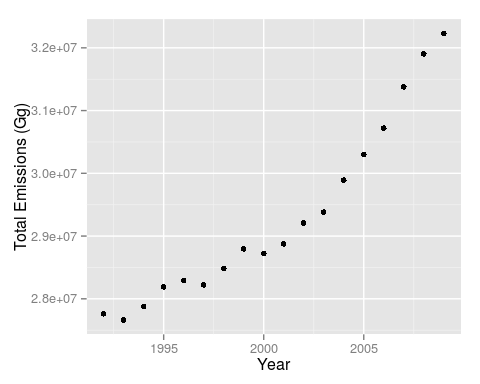
## Aggregation Example using dplyr: plot averages by year  
emissionsDataFAOSTAT %>%  
 group\_by(Year) %>%  
 summarize(totalEmissions = sum(Emissions, na.rm = TRUE))

## Source: local data frame [55 x 2]  
##   
## Year totalEmissions  
## 1 1961 16609671  
## 2 1962 16962077  
## 3 1963 17187564  
## 4 1964 17574667  
## 5 1965 17944599  
## 6 1966 18400841  
## 7 1967 18838752  
## 8 1968 19193044  
## 9 1969 19392537  
## 10 1970 19667815  
## .. ... ...

emissionsDataFAOSTAT %>%  
 group\_by(Year) %>%  
 summarize(totalEmissions = sum(Emissions, na.rm = TRUE)) %>%  
 qplot(Year, totalEmissions, data = .) + labs(y = "Total Emissions (Gg)")



emissionsDataFAOSTAT %>%  
 filter(Year > 1991 & Year < 2010) %>%  
 group\_by(Year) %>%  
 summarize(totalEmissions = sum(Emissions, na.rm = TRUE)) %>%  
 qplot(Year, totalEmissions, data = .) + labs(y = "Total Emissions (Gg)")



emissionsDataFAOSTAT %>%  
 filter(Year > 1990 & Year < 2010) %>%  
 group\_by(FAOST\_CODE) %>%  
 summarize(meanEmissions = mean(Emissions, na.rm = TRUE)) %>%  
 arrange(meanEmissions)

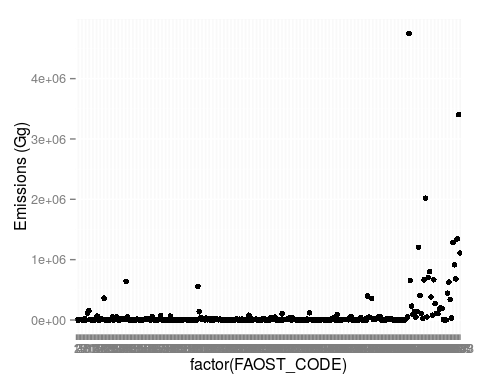
## Source: local data frame [279 x 2]  
##   
## FAOST\_CODE meanEmissions  
## 1 6 0  
## 2 22 0  
## 3 82 0  
## 4 94 0  
## 5 127 0  
## 6 140 0  
## 7 161 0  
## 8 163 0  
## 9 172 0  
## 10 180 0  
## .. ... ...

emissionsDataFAOSTAT %>%  
 filter(Year > 1990 & Year < 2010) %>%  
 group\_by(FAOST\_CODE) %>%  
 summarize(meanEmissions = mean(Emissions, na.rm = TRUE)) %>%  
 arrange(desc(meanEmissions))

## Source: local data frame [279 x 2]  
##   
## FAOST\_CODE meanEmissions  
## 1 5000 4748377.6  
## 2 5849 3404747.8  
## 3 5300 2017661.6  
## 4 5848 1343213.5  
## 5 5815 1285542.0  
## 6 5200 1204559.6  
## 7 5873 1111912.1  
## 8 5817 917373.2  
## 9 5303 802278.3  
## 10 5302 708064.6  
## .. ... ...

emissionsDataFAOSTAT %>%  
 filter(Year > 1990 & Year < 2010) %>%  
 group\_by(FAOST\_CODE) %>%  
 summarize(meanEmissions = mean(Emissions, na.rm = TRUE)) %>%  
 arrange(desc(meanEmissions)) %>%  
 qplot(data = ., x = factor(FAOST\_CODE), y = meanEmissions) +  
 labs(y = "Emissions (Gg)")

## Warning: Removed 1 rows containing missing values (geom\_point).



emissionsDataFAOSTAT %>%  
 filter(Year > 1990 & Year < 2010) %>%  
 group\_by(FAOST\_CODE) %>%  
 summarize(meanEmissions = mean(Emissions, na.rm = TRUE)) %>%  
 arrange(desc(meanEmissions)) %>%  
 filter(meanEmissions >= 200000) %>%  
 qplot(data = ., x = factor(FAOST\_CODE), y = meanEmissions) +  
 labs(y = "Emissions (Gg)")

